06-05-03

AF/9700$

PTO/SB/21 (01-03)
Approved for use through 04/30/2003. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Application Number	09/405,618 confirmation # 9689
Filing Date	09/24/1999
First Named Inventor	Blomgren, et al
Art Unit	2123
Examiner Name	Craig, Dwin M.
Attorney Docket Number	31876.0140

Total Number of Pages in This Submission

ENCLOSURES (*Check all that apply*)

- [x] Fee Transmittal Form
  - [ ] Fee Attached
- [ ] Amendment/Reply
  - [ ] After Final
  - [ ] Affidavits/declaration(s)
- [ ] Extension of Time Request
- [ ] Express Abandonment Request
- [ ] Information Disclosure Statement
- [ ] Certified Copy of Priority Document(s)
- [ ] Response to Missing Parts/ Incomplete Application
  - [ ] Response to Missing Parts under 37 CFR 1.52 or 1.53
- [ ] Drawing(s)
- [ ] Licensing-related Papers
- [ ] Petition
- [ ] Petition to Convert to a Provisional Application
- [ ] Power of Attorney, Revocation Change of Correspondence Address
- [ ] Terminal Disclaimer
- [ ] Request for Refund
- [ ] CD, Number of CD(s) _____
- [ ] After Allowance Communication to Group
- [x] Appeal Communication to Board of Appeals and Interferences
- [ ] Appeal Communication to Group **(Appeal Notice, Brief, Reply Brief)**
- [ ] Proprietary Information
- [ ] Status Letter
- [x] Other Enclosure(s) (please Identify below):

Remarks

postcard

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm or Individual	Matthew J. Booth, Reg. No. 35,454	Booth & Wright, LLP P. O. Box 50010 Austin, TX 78763-0010
Signature	Matthew J. Booth	
Date	06/04/2003	

CERTIFICATE OF TRANSMISSION/MAILING

I hereby certify that this correspondence is being facsimile transmitted to the USPTO or deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, Washington, DC 20231 on this date: 06/04/2003

Typed or printed	Matthew J. Booth		
Signature	Matthew J. Booth	Date	06/04/2003

This collection of information is required by 37 CFR 1.5. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, Washington, DC 20231.**

If you need assistance in completing the form, call 1-800-PTO-9199 (1-800-786-9199) and select option 2.

PTO/SB/17 (01-03)
Approved for use through 04/30/2003. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

FEE TRANSMITTAL for FY 2003

Effective 01/01/2003. Patent fees are subject to annual revision.

☐ Applicant claims small entity status. See 37 CFR 1.27

TOTAL AMOUNT OF PAYMENT ($) 320.00

Complete if Known	
Application Number	09/405,618
Filing Date	09/24/1999
First Named Inventor	Blomgren, et al.
Examiner Name	Craig, Dwin M.
Art Unit	2123 confirmation #9689
Attorney Docket No.	31876.0140

METHOD OF PAYMENT *(check all that apply)*

☐ Check ☐ Credit card ☐ Money Order ☐ Other ☐ None

☑ Deposit Account:

Deposit Account Number: 11-0851

Deposit Account Name: Booth & Wright, LLP

The Commissioner is authorized to: *(check all that apply)*

☑ Charge fee(s) indicated below ☑ Credit any overpayments

☑ Charge any additional fee(s) during the pendency of this application

☐ Charge fee(s) indicated below, **except for the filing fee** to the above-identified deposit account.

FEE CALCULATION

1. BASIC FILING FEE

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
1001	750	2001	375	Utility filing fee	
1002	330	2002	165	Design filing fee	
1003	520	2003	260	Plant filing fee	
1004	750	2004	375	Reissue filing fee	
1005	160	2005	80	Provisional filing fee	

SUBTOTAL (1) ($)

2. EXTRA CLAIM FEES FOR UTILITY AND REISSUE

		Extra Claims		Fee from below	Fee Paid
Total Claims		-20** =		X	=
Independent Claims		- 3** =		X	=
Multiple Dependent					=

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description
1202	18	2202	9	Claims in excess of 20
1201	84	2201	42	Independent claims in excess of 3
1203	280	2203	140	Multiple dependent claim, if not paid
1204	84	2204	42	** Reissue independent claims over original patent
1205	18	2205	9	** Reissue claims in excess of 20 and over original patent

SUBTOTAL (2) ($)

**or number previously paid, if greater; For Reissues, see above

FEE CALCULATION (continued)

3. ADDITIONAL FEES

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
1051	130	2051	65	Surcharge - late filing fee or oath	
1052	50	2052	25	Surcharge - late provisional filing fee or cover sheet	
1053	130	1053	130	Non-English specification	
1812	2,520	1812	2,520	For filing a request for *ex parte* reexamination	
1804	920*	1804	920*	Requesting publication of SIR prior to Examiner action	
1805	1,840*	1805	1,840*	Requesting publication of SIR after Examiner action	
1251	110	2251	55	Extension for reply within first month	
1252	410	2252	205	Extension for reply within second month	
1253	930	2253	465	Extension for reply within third month	
1254	1,450	2254	725	Extension for reply within fourth month	
1255	1,970	2255	985	Extension for reply within fifth month	
1401	320	2401	160	Notice of Appeal	
1402	320	2402	160	Filing a brief in support of an appeal	$320.00
1403	280	2403	140	Request for oral hearing	
1451	1,510	1451	1,510	Petition to institute a public use proceeding	
1452	110	2452	55	Petition to revive - unavoidable	
1453	1,300	2453	650	Petition to revive - unintentional	
1501	1,300	2501	650	Utility issue fee (or reissue)	
1502	470	2502	235	Design issue fee	
1503	630	2503	315	Plant issue fee	
1460	130	1460	130	Petitions to the Commissioner	
1807	50	1807	50	Processing fee under 37 CFR 1.17(q)	
1806	180	1806	180	Submission of Information Disclosure Stmt	
8021	40	8021	40	Recording each patent assignment per property (times number of properties)	
1809	750	2809	375	Filing a submission after final rejection (37 CFR 1.129(a))	
1810	750	2810	375	For each additional invention to be examined (37 CFR 1.129(b))	
1801	750	2801	375	Request for Continued Examination (RCE)	
1802	900	1802	900	Request for expedited examination of a design application	

Other fee (specify) _____

*Reduced by Basic Filing Fee Paid

SUBTOTAL (3) ($) 320.00

SUBMITTED BY

(Complete *(if applicable)*)

Name *(Print/Type)*	Matthew J. Booth	Registration No. *(Attorney/Agent)*	35,454	Telephone	512-474-8488
Signature	Matthew J. Booth			Date	06/04/2003

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

This collection of information is required by 37 CFR 1.17 and 1.27. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, Washington, DC 20231.**

If you need assistance in completing the form, call 1-800-PTO-9199 (1-800-786-9199) and select option 2.

In The United States Patent & Trademark Office
Before The Board Of Patent Appeals And Interferences

Inventor(s):	Blomgren et al
Serial No.:	09/405,618
Confirmation No.	9689
Filed:	24.09.1999 (24 September 1999)
For:	Software Modeling of Logic Signals Capable of Holding More than Two Values
Docket No.:	31876.0140
Art Unit:	2123
Examiner:	Craig, Dwin M.

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Appeal Brief

Real Party in Interest

Intrinsity, Inc., formerly EVSX, Inc., of Austin, Texas is the real party in interest in this appeal. Intrinsity filed a change of its corporate name with the USPTO on 23 May 2000.

Related Appeals and Interferences

Applicant identifies Applicant's Appeal of the Examiner's Final Rejection of Claims 1-25 in U.S. Patent Application Serno. 09/406,017, entitled "Method And Apparatus For A Monitor That Detects and Reports A Status Event To A Database," filed on 24 Sept. 1999 (Notice of Appeal filed on 4 April, 2003) as potentially being related to this Appeal. While the Board's decisions in that case and in this case may not directly affect or have bearing on each other, the two cases have the same Examiner

and the same art (*Giramma*, U.S. Pat. No. 5,706,476) is being cited by the Examiner as supporting his rejections in both cases.

Status of Claims

Applicant is appealing from the final rejection of claims 1-3, 6-8, 11-13,16-18, and 21-23 that the Examiner made in the Final Office Action of 10 January 2003. Appendix A contains the currently pending claims.

Status of Amendments

Applicant has not filed any amendment subsequent to the Final Rejection.

Summary of the Invention

The claimed invention is a method and apparatus that enables the efficient simulation of logic designs comprising nonbinary signals that are capable of having more than two unique decimal values and multiple unique drive states, such as designs based upon the novel NDL logic design style. P. 9, lines 2-4. Intrinsity Inc. (f/k/a EVSX Inc.), the assignee of this application, has developed a new dynamic logic design style and an associated family of semiconductor devices, originally designated as N-NARY logic, and now known as NDL logic implemented in FAST14 technology. N-NARY dynamic logic circuits (denoted as "NDL gates") comprise a logic tree circuit that couples to one or more input logic paths and one or more output logic paths. The logic tree circuit is a single, shared logic tree comprising transistors organized into multiple evaluation paths that evaluate the function of the logic circuit (e.g., an AND/NAND function, an OR/NOR function, or an XOR/Equivalence function, and the like). The logic

tree is precharged using a precharge circuit and evaluates using an evaluate circuit, both of which are controlled by a clock signal. P. 2, lines 9-12; FIG. 6.

The input and output logic paths in a NDL gate, and indeed, the datapath between successive NDL gates typically do not carry binary signals. P. 8, lines 4-12; P. 12 lines 14-17. Instead, NDL logic uses a new signal structure, a "1-of-N signal", which is a multi-wire 1-hot signal that conveys a signal value using a specifically defined encoding scheme. P. 7, line 4- P. 8, line 12. Intrinsity's NDL logic and 1-of-N signal technology is illustrated and described in US Pat. No. 6,069,497, entitled "Method and Apparatus for a N-NARY Logic Circuit Using 1 of N Signals", which is incorporated by reference for all purposes into the specification disclosing the instant invention and denoted herein as "the NDL Patent."[1] P. 2, lines 9-12. As described in the NDL Patent, NDL logic and 1-of-N signals can be used to build adders, shifters, multipliers, boolean units, RAM devices, and even entire datapaths. Id. The NDL Patent also describes the encoding scheme used in multi-wire 1-of-N signals, where the 1-of-N signal's decimal value is encoded according to *which specific* wire of the multi-wire signal bundle is asserted while all other wires in the multi-wire signal bundle are not asserted. Id. That description is repeated in the instant application, and illustrated using a 1-of-4 signal as an example. P. 7, line 4- P. 8, line 12. As described in the instant specification, a 1-of-4 signal is composed of four related wires and has four possible decimal values 0-3. Id. The decimal value of the 1-of-4 signal is dictated by which one, and only one, of the

[1] Since the instant application was filed, the USPTO has issued 49 patents disclosing and claiming NDL-related inventions. A listing of patents assigned to Intrinsity that relate to Intrinsity's new NDL logic family appears at Appendix B.

four signal wires is asserted, while the other three wires in the multi-wire bundle are not asserted. Id.

The present invention models 1-of-N signals for simulation purposes in a specific format comprising a signal value field that holds the signal's unique decimal value that corresponds solely to the logic value encoded in the signal's N wires, a signal strength field that conveys the drive state of the signal, which is a wholly different characteristic of the 1-of-N signal than its logic value, and a signal definition field that conveys whether the 1-of-N signal is a defined or an undefined signal. P. 9, lines 4-7; P. 19, lines 4-15; FIG. 4; P. 24 line 3-p. 26, line 12. The unique decimal value of a defined signal occupies the least significant bits in the model, thus allowing the simulator to perform mathematical and logical operations on the gate's input and output signals at the signal value level, rather than at the wire/bit level, as is done in prior art extended-state simulation models. P. 19, lines 4-15; FIG. 4, P. 21, lines 1-3; P. 21, line 14-P. 22, line 3; P. 26, line 13-P. 27, line 15.

Issues

The primary issue on appeal is whether or not claims 1-3, 6-8, 11-13,16-18, and 21-23 are anticipated under 35 USC 102(b) by *Giramma* (US Pat. No. 5,706,476), by the *IEEE Standard Multivalue Logic System for VHDL Model Interoperability* (Std_logic_1164), and by *On the Use of VHDL as a Multi-Valued Logic Simulator* by C. Rozon, IEEE 1996. In addition, Applicant seeks the Board's review of the Examiner's rejection of claims 1-3, 6-8, 11-13,16-18, and 21-23 as being unpatentable under the

judicially created doctrine of obviousness-type double patenting in view of Leight et al., US Pat. No. 6,289,497.

Grouping of Claims

The groupings of claims 1-3, 6-8, 11-13, 16-18, and 21-23 will stand or fall together. For the purposes of this appeal, Applicant will discuss claims 1-3 as illustrative of claims 6-8, 11-13,16-18, and 21-23.

The Examiner's Rationale

The Examiner's rationale for rejecting claims 1-3, 6-8, 11-13,16-18, and 21-23 as anticipated by *Giramma* was stated as follows in his final rejection:

> *Giramma* discloses, taking as per claim(s) 1, 6, 11, 16, and 21 for example: A model that simulates logic signals capable of having more than two unique values and one or more unique drive states. **Col. 1, lines 20-63** and in claim 1 *Giramma* recites, "A computer-assisted logic gate simulation method for simulating a circuit that includes plural logic gates characterized by differing output state abstractions," whereby the unique driver states comprise: a signal value field *Giramma* discloses, "Those of skill in the art will appreciate that by directive production is meant any technique for signaling or otherwise communicating to a downstream connected primitive the possible next-state of its inputs based upon a transition of state of an output of an upstream primitive." **Col. 5, lines 13-18.** *Giramma* recites a signal strength field, "In addition to these state abstractions, some logic simulators use a 3-state model coupled with essentially unlimited strengths or values." **Col. 1, lines 51-53**. *Giramma* also teaches a signal definition field which comprises information that conveys whether the signal being modeled holds a defined value or an undefined value, "Also preferably, the 4-state abstraction includes an undefined state such as undefined state XS described above." **Col. 4, lines 53-55**.
>
> With respect to claims 2, 7, 12, 17, and 22 *Giramma* discloses, "Those skilled in the art will appreciate that 32-bit 'zoom' words, in accordance with the preferred embodiment of the invention, are used to obtain indices into various tables for next-state logic evaluations." **Col. 6, Lines 51-54**.
>
> With respect to claims 3, 8, 13, 18, and 23 *Giramma* discloses, "Multiple-Value Logic 9-state model (MVL-9) of 0, 1, X at strong and

> resistive strengths or values, as well as high-impedance uninitialized, and undefined (0S, 1S, XS, 0R, 1R, XR, Z, U, D)." **Col. 1, Lines 46-49**. [sic, emphasis in original].

The Examiner's rationale for rejecting claims 1-3, 6-8, 11-13, 16-18 and 21-23 under 35 U.S.C. 102(b) as being anticipated by the IEEE Standard Multivalue Logic System for VHDL Model Interoperability (Std_logic_1164) was stated as follows:

> *IEEE Standard Multivalue Logic System for VHDL Model Interoperability* discloses, with respect to claim(s) 1, 6, 11, 16 and 21, logic signals capable of having more than two unique values and one or more unique drive states. On Page 2, *IEEE Standard Multivalue Logic System for VHDL Model Interoperability* recites, SUBTYPE UXO1Z IS resolved std_ulogic RANGE 'U' TO 'Z'; -- ('U','X','O','1','Z'). *IEEE Standard Multivalue Logic System for VHDL Model Interoperability* recites on in **Annex A on Page 15**, "A.1 Value system... one must interpret the meaning of each of the elements as provided by the standard. Type std_ulogic is ('U', Uninitialized state Used as a default value 'X', Forcing Unknown Bus contentions, error conditions, etc. '0', Forcing Zero Transistor driven to GND 'I' Forcing One Transistor driven to VCC 'Z', High Impedance 3-state buffer outputs 'W', Weak Unknown Bus terminators 'L', Weak Zero Pull down resistors 'H', Weak One Pull up resistors '-' Don't Care Used for synthesis and advanced modeling);" which is a signal value field in the Multivalue Logic System as well as a signal definition field. *IEEE Standard Multivalue Logic System for VHDL Model Interoperability* teaches on **Page 15, Annex A**; "Therefore, a number of strength stripper functions have been designed to transform 'Z', 'W', 'L', 'H', and '-' into their corresponding forcing strength counterparts." This teaches a signal strength field.
>
> With respect to claims 2, 3, 7, 8, 12, 13, 17, 18, 22 and 23 *IEEE Standard Multivalue Logic System for VHDL Model Interoperability* teaches on **Page 15, Annex A**, "Forcing Zero Transistor driven to GND, Forcing One Transistor driven to VCC, Weak Zero Pull Down resistors, Weak One Pull up Resistors. [sic, emphasis in original].

The Examiner's rationale for rejecting claims 1-3, 6-8, 11-13, 16-18 and 21-23 under 35 U.S.C. 102(b) as being anticipated by the *On the Use of VHDL as a Multi-Valued Logic Simulator* by C. Rozon, IEEE 1996 (hereinafter, *Rozon*) was stated as follows:

As regards **Claims 1,6,11,16 and 21**, the *Rozon* reference teaches a signal model for an N-Nary logic simulation **Pages 110-115**. wherein the logic value comprises an integer greater than 1 Table 1, page 110, and a signal strength **Table 1, page 110** and signal definition **Page 112, and Table 2.**

As regards **Claims 2,7,12,17 and 22** the *Rozon* reference discloses an integer less than or equal to 31 **ANNEX 1, page 114**.

As regards **Claims 3,8,13,18 and 23** the *Rozon* reference discloses the signal strength being high-impedance, weakly-driven, moderately driven or strongly-driven, **Page 110 Table 1**. [sic, emphasis in original].

Argument

Intrinsity Inc. (f/k/a EVSX Inc.), the assignee and the real party in interest of this application, has developed a new dynamic logic design style and an associated family of semiconductor devices, originally designated as N-NARY logic, and now known as NDL logic implemented in FAST14 technology. N-NARY dynamic logic circuits (denoted as "NDL gates") are interconnected using a new signal structure, a "1-of-N signal", which is a multi-wire 1-hot signal that conveys a decimal signal value using a specifically defined encoding scheme. A listing of patents assigned to Intrinsity that relate to Intrinsity's new NDL logic family appears at Appendix B, which is current through 27 May 2003. The present invention is a signal model used to simulate NDL logic that supports simulating various characteristics of multiwire, 1-hot 1-of-N signals including the signal's nonbinary decimal value, its drive strength, and its defined/undefined or initialized/uninitialized status.

35 U.S.C. § 102(b) Rejections.

Despite Applicant's best efforts to explain and illustrate, the Examiner has never developed an understanding of the substantial differences between the Assignee's nonbinary FAST14 technology and modeling thereof using the present invention and

standard, well-known extended-state modeling techniques applicable to binary logic. In the First Office Action, the Examiner erroneously rejected the original claims presented as being anticipated by *Giramma* and the IEEE Specification Std_logic_1164, both of which disclose extended-state modeling techniques well known by those skilled in the art to be applicable to binary logic. In response, Applicant pointed out the Examiner's error and failure to understand the substantial differences between extended-state modeling for binary logic and the extended state modeling techniques for NDL logic disclosed and claimed in the present invention. Applicant also amended the claims to more particularly point out and distinctly claim the applicability of the present invention to 1-of-N signals used in NDL logic. Applicant's Amendment in Response to the First Office Action, filed on 16 October 2002, is incorporated herein by reference for all purposes.

Nevertheless, Examiner maintained his erroneous rejections and repeated them in the Final Office Action, not even bothering to alter his rationale to reflect the amended claim language. In addition, the Examiner added a new rejection based upon a new piece of art, the *Rozon* article, which he contends teaches a signal model for an N-NARY logic simulation. Consequently, the Examiner has erroneously rejected claims 1-3, 6-8, 11-13,16-18, and 21-23 as being anticipated by three separate pieces of prior art (*Giramma*, IEEE Specification Std_logic_1164, and *Rozon*). As discussed in Applicant's response to the First Office Action, *Giramma* and the IEEE specification disclose extended-state modeling techniques well known by those skilled in the art to be applicable to binary logic. As discussed further herein, *Rozon* explores how binary-based VHDL simulation techniques that account for multiple drive states of binary

signals might be extended to non-binary (i.e., multi-valued) logic simulations and shows the simulation of a ternary (radix-3) full adder. However, other than showing how a ternary adder might be simulated, *Rozon* does not disclose or enable the simulation of other common functions that designers would use as the "building blocks" of a ternary (or other multi-valued) logic system.[2] Consequently, Rozon's article does not disclose a simulation methodology for non-binary logic, rather, it is simply a very high level discussion of how one might go about developing a simulation methodology for non-binary logic ("establish an algebra", "choose a compiler", decide on a description style, etc.) *Rozon* also contends that adding IEEE 1164 simulation extensions such as drive states and don't care values to a non-binary simulation methodology can only be done "at the cost of increasing the model's complexity." P. 111, first column. In contrast, the present invention discloses a simulation model and methodology for non-binary logic that includes the "complete algebra" to enable the simulation that was out of the scope of *Rozon's* article, and also accounts for a number of IEEE 1164 simulation states, including drive state, without increasing the model's complexity.

As discussed in detail in Applicant's Response to the First Office Action, *Giramma* discloses a methodology by which a binary-based logic design can be simulated using a combination of an 8-state simulation model and a 4-state simulation model. *Giramma's* 8-state model is a modification of the standard, well-known MVL-9 model discussed in the instant application and in the IEEE Specification Std_logic_1164, and his 4-state model is a modification of the well-known prior art 4-

[2] Indeed, *Rozon* briefly discusses the dramatic increase in the complexity of the simulation model that occurs when simulating ternary logic using VHDL techniques but states that developing the operators needed to create the functionally complete algebra that is necessary to simulate ternary logic "is out of the scope of this paper." P. 111, first column.

state model discussed in the instant application. Like the discussion in the current application at page 5, line 14 through page 6, line 20, *Giramma* discusses the impracticality of using extended-state models, such as an 8- or 9-state simulation model, to simulate complex binary gates, because the size of the truth-tables involved cannot be accommodated in the memory of a standard computer simulation workstation. The present application discloses that designers attempting to employ extended-state simulations for complex binary logic have developed workaround solutions, such as mapping large truth table functions to multiple smaller truth table functions, or ignoring certain states within a multiple state model for certain designs. Application at p. 6, lines 16-20. In fact, *Giramma's* method is a prime example of the prior art workaround solutions discussed in the present application, because *Giramma* has combined both approaches to produce a more efficient binary logic simulation methodology than a straight extended-state approach. In a nutshell, *Giramma* employs an extended 8-state model only on certain gates in a logic path. For all other gates, *Giramma* maps certain states within the 8-state model to one of the states of the 4-state model, and then simulates the gates' outputs using the modified and mapped 4-state model. *Giramma* has redefined the standard 4-state model to ignore the high-impedance (Z) state, and has modified the MVL-9 model disclosed in the IEEE 1164 specification to ignore the undefined (D) state. Therefore, *Giramma's* invention both ignores certain states of the standard models, and maps states of the extended-state model to states of a lower-state model in order to use a smaller truth table to determine a gate's output. This is precisely the kind of prior art discussed in the instant application that is only applicable to binary-based logic designs. As discussed in the instant

application, using this kind of approach to simulate even a few gates of an N-NARY design is virtually impossible. The present invention is thus an entirely new method and apparatus that enables designers to simulate *nonbinary* logic, such as N-NARY logic. As such, it is very different from prior art methods like *Giramma's* invention or the standard extended-state models discussed in the IEEE Specification.

Rozon's article explores, at a very high level, adapting VHDL logic simulation techniques typically used in simulations of binary, multiple drive state logic to non-binary logic signals that may also be uninitialized, unknown, or high-impedance, but *Rozon does not* include accounting for different drive strengths (i.e., strongly-driven, moderately-driven, weakly-driven) of his non-binary signals. *Rozon* presents a 5-step approach to adapting standard binary logic simulation techniques to develop a non-binary logic simulation methodology, but Rozon's method steps are described at *such* a high level ("establish an algebra"; "select and use an existing compiler"; "decide on the description style"; and so on) that, unlike the present invention, no *specific* implementation is disclosed.

Before turning to an analysis of the three references as applied to the elements of each rejected claim, a discussion regarding terminology is required. Applicant believes that the Examiner's confusion likely stems from terminology. In Applicant's lexicon as applied to a binary signal, the phrases "logic value" and "decimal value" refer ***only*** to the true-or-false, high-or-low, 1-of-0 nature of the signal—the information used to determine whether a binary signal is asserted or not. In Applicant's lexicon, a signal's "logic value" or "decimal value" bears no relation to its drive state or whether the signal has a status other than asserted or not asserted (i.e., whether a signal is undefined,

uninitialized, or high-impedance). The signal model of the present invention is a representation of a 1-of-N signal that is in a state that includes its logic or decimal value, its signal strength, and its signal status—all three of which are separate pieces of information about the signal.

Using Applicant's lexicon, then, *every* prior art simulation model used for binary logic, whether it employs the simplest 2-state logic model or the most complex extended-state model, can only simulate signals that have a "decimal value" or "logic value" of 1 or 0, regardless of the terminology used that may confuse a signal's "logic value" or "decimal value" with the number of states that the model can represent. In the more sophisticated models, as described in the instant application and in *Giramma*, these binary signals having a decimal value or logic value of either 1 or 0 also have a "drive state," which may be either strongly driven (typically denoted as S) or weakly-driven (typically denoted as R). *Notably, a strongly-driven high-voltage binary signal does* ***not*** *have a different signal value, decimal value, or logic value than a weakly driven high-voltage binary signal; both signals are asserted and have a logic value of 1. They simply have different drive strengths.* Similarly, like a signal's drive strength, prior art simulation models can simulate other characteristics of signals that are not the same as the signal's logic value—for instance, a signal with unknown or indeterminate value (X), a high-impedance signal (Z), an uninitialized signal (U), or an undefined signal (D). In Applicant's lexicon, these kinds of signals have no logic or decimal value—they are unknown, high-impedance, uninitialized, or undefined. These are not "decimal values," they are signal characteristics that can be meaningfully simulated.

Unfortunately, despite this explanation, the Examiner continues to mischaracterize the ability of prior art models to simulate various signal characteristics of binary signals—signals limited to having a logic or decimal value of only 1 or 0—as an ability to simulate 1-of-N signals having logic values other than 1 or 0—i.e., 1-of-N signals used in NDL logic. For example, in the Final Office Action at 3, the Examiner concludes that "the IEEE reference discloses an N-NARY signal model where "N" is greater than 2" simply because page 15, Annex A of the IEEE reference shows a model capable of accounting for 9 different states of a binary signal. The number of signal states that a signal model can represent bears absolutely no relation to the "N" value of a 1-of-N signal as used in NDL logic. The Examiner's assertion demonstrates a fundamental lack of understanding of the difference between standard, prior art extended-state modeling techniques for binary logic and the present invention.

The multiple state representation that appears on page 15, Annex A of the IEEE reference to which the Examiner refers is actually the MVL-9 state model described in the instant application at p. 4, lines 18-21 and Table 1. For convenience, the states modeled in IEEE reference (described in a VHDL variable type declaration on p. 15, Annex A) are repeated below:

```
Type std_ulogic is (
        'U' ,          Uninitialized state     Used as a default value
        'X' ,          Forcing Unknown         Bus contentions, error conditions, etc.
        '0' ,          Forcing Zero            Transistor driven to GND
        '1' ,          Forcing One             Transistor driven to VCC
        'Z' ,          High Impedance          3-state buffer outputs
        'W' ,          Weak Unknown            Bus terminators
        'L' ,          Weak Zero               Pull down resistors
        H' ,           Weak One                Pull up resistors
        '-' ,          Don't Care              Used for synthesis and advanced modeling
);
```

Likewise, Table 1 on p. 5 of the instant application is repeated below for convenience:

Table 1-Prior Art

Model state level	Logic states supported	Definitions
2-state	0, 1	
3-state	0, 1, X	X=uninitialized
4-state	0, 1, X, Z	Z="undriven" or high-impedance
9-state	0S, 1S, XS, 0R, 1R, XR, 0Z, 1Z, XZ	S="Strongly" driven; R="weakly" driven; Z=not driven, value of wire=last known value
MVL-9	0S, 1S, XS, 0R, 1R, XR, Z, UI, UD	Z=not driven; last value of wire ignored UI=uninitialized; last value of wire ignored UD=undefined; last value of wire ignored
12-state	0S, 1S, XS, 0R, 1R, XR, 0Z, 1Z, XZ, 0I, 1I, XI	I=Indeterminate

A simple comparison reveals that the model described in the IEEE reference is, in fact, the MVL-9 state model described in Table 1 of the instant application and the text at p. 3, line 19-p. 4, line 21. The state designated as '0' in the IEEE spec corresponds to the state described as '0S' in the MVL-9 model line of Table 1 above; it represents a strongly driven binary signal having a logic value of '0'. Similarly, the state designated as the 'L' state in the IEEE reference corresponds to "0R" in Table 1, representing a weakly driven binary signal having a logic value of '0'. Continuing on, 'X' corresponds to 'XS'; 'W' corresponds to 'XR'; 'U' corresponds to 'UI' and '-' corresponds to 'UD'. As described in the instant application and in the IEEE

specification, the "strong" and "resistive" states do not represent a different logic value, they simply provide a way to model some of the physical aspects of the circuitry. As described in the instant application at p. 4, l. 14-15, the S state indicates the value is driven "strong", usually by having an ON transistor connected to the appropriate power or ground plane. The R state indicates that the value is driven "weak", usually by having a resistor (often constructed from a properly biased transistor) connected to the appropriate power or ground plane. This is reflected in the IEEE specification by the notations "transistor driven to GND"; "transistor driven to VCC" (strongly driven signals); and "pull up resistors" and "pull down resistors" (weakly driven signals).

The Examiner refuses to recognize the distinction between a signal's logic value and its drive state. For example, the Examiner "asserts that a logic state of "1" does describe a drive state and that a logic state of "0" describes another drive state. Final Office Action at 5. While that may be true as an abstract principle, Applicant has clearly defined and used the term "drive state" to refer only to the strength of a signal, independent of its value.[3] Likewise, Applicant has clearly explained that a 1-of-N signal's "value" means the value encoded in the 1-of-N signal's N wires, as determined by which one (and only one) of the N wires within the bundle is asserted. P. 7, line 6-10. In Applicant's lexicon, a signal's drive state and its logic value are two distinctly different things, and those skilled in the art would not misunderstand Applicant's use of

[3] Applicant has clearly defined a strongly driven signal to mean a signal ordinarily created via a connection through a conducting transistor directly to VCC or to ground. P. 4 lines 14-15. Likewise, at p. 4, lines 15-17, Applicant has clearly defined a weakly driven signal to mean a signal ordinarily created by having a resistor (often constructed from a properly biased transistor) connected to the appropriate power or ground plane. Obviously, the signal's logic value will be different, depending upon whether it is connected to VCC or to ground, but its drive strength will not change because, in Applicant's lexicon, the "drive strength" of a signal is dictated by the physical layout of the circuit being simulated. This is completely consistent with the common usage and understanding of those skilled in the art.

the two terms or their meaning as used in the specification disclosing the instant invention and in the claims. By failing to recognize and give meaning to the distinction, the Examiner failed to give the claims the broadest reasonable interpretation consistent with the specification and consistent with the interpretation that those skilled in the art would reach, as required by MPEP 2111.

With this understanding, then, it is clear that none of the three references cited by the Examiner anticipates the present invention under 35 U.S.C. §102. To be a proper §102 reference, the reference must show each and every element and limitation of the claimed invention. The MPEP provides the following with regard to anticipation of a claim: "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference. <omitted citations> The identical invention must be shown in as complete detail as is contained in ... the claim. <omitted citations>." MPEP § 2131. Additionally, the MPEP further provides: "In other words, for anticipation under 35 USC 102, the reference must teach every aspect of the claimed invention either explicitly or impliedly. Any feature not directly taught must be inherently present." MPEP § 706.02. *Giramma*, the IEEE Specification Std_logic_1164, and *Rozon* are not proper §102 references because none of these references teaches or shows all of the elements and limitations of the present invention, as discussed in more detail below. The following discussion address the Examiner's rationale for rejection, applying each reference individually to Claims 1, 2, and 3.

Claim 1: *A signal model used in an N-NARY logic simulation, comprising:*
a signal value, said signal value further comprises the logic value of a nonbinary 1-of-N logic signal being modeled, wherein said logic value further comprises an integer greater than 1;

a signal strength, said signal strength further comprises the drive state of said nonbinary 1-of-N logic signal being modeled; and

a signal definition, said signal definition further comprises the defined or undefined status of said nonbinary 1-of-N logic signal being modeled.

Giramma. Since the Examiner maintained his §102(b) rejections of Claim 1, but failed to explain how the cited art teaches every element and every limitation of the claim as amended, Applicants are left somewhat in the dark as to the Examiner's rationale as applied to the new claim language. The Examiner apparently contends that in *Giramma*, Col. 1, lines 20-63, Col. 5, lines 13-18, Col. 1, lines 51-53, and Col. 4, lines 53-55 collectively teach every element and every limitation of Claim 1. It is true that *Giramma* teaches a signal modeling methodology that includes a value, a strength, and defined/undefined states, just like the prior art extended-state signal models used to simulate binary logic discussed at length in the background material in the instant specification. However, as the Examiner has failed to understand, a model capable of modeling different states of a binary signal, such as Giramma's model, is not the same as the model of the present invention, because the present invention models multiwire 1-of-N signals, which have very different physical and functional characteristics than binary signals.

The Examiner states that Col. 1, lines 20-63 and Giramma's claim 1, which states "A computer-assisted logic gate simulation method for simulating a circuit that includes plural logic gates characterized by differing output state abstractions" teaches the signal model recited in the preamble. Of course, Applicants are not disputing that *Giramma* is teaching a signal model. However, *Giramma* is teaching a signal model and simulation methodology to model the multiple states of binary signals. The claim preamble recites

a signal model used in an N-NARY logic simulation. Notwithstanding, this is not worth arguing about because the preamble is simply reciting the intended use of the claimed signal model; the claim does not depend upon the preamble because the claim elements stand on their own. Accordingly, the preamble is not accorded patentable weight. MPEP 2111.02.

The Examiner apparently contends that the signal value element claimed in Claim 1 is taught by the following statement: "Those of skill in the art will appreciate that by directive production is meant any technique for signaling or otherwise communicating to a downstream connected primitive the possible next-state of its inputs based upon a transition of state of an output of an upstream primitive." *Giramma*, Col. 5, lines 13-18. In Giramma's lexicon, providing a "downstream directive" means to propagate the output of one gate to the input of a downstream gate, passing along signal characteristics (like signal value and drive state) according to whatever propagation rules the user has selected. Applicant is unsure how or why the Examiner picked this statement as teaching a signal value, however, Applicant does not dispute that *Giramma* teaches propagating a modeled signal that has a signal value. However, neither this statement, nor any other teaching in *Giramma* teaches the additional functional limitations included in the claimed signal value; namely, that it is the logic value of a nonbinary 1-of-N logic signal being modeled, wherein said logic value further comprises an integer greater than 1. Moreover, there is nothing in *Giramma* that teaches the modeling or use of 1-of-N logic signals. 1-of-N logic signals are specifically defined in the specification to be multiwire signals comprised of N wires, and capable of holding N values, wherein one and only one wire will be asserted for a valid signal. p. 7,

line 4-p. 8, line 3. *Giramma* does not describe his invention in terms of modeling or simulating any type signal having these very specific claimed characteristics, because Giramma's invention only models and simulates binary, single-wire signals. As discussed in detail above, modeling several states of a binary signal (which states may include its possible logic value of 0 or 1, along with other signal characteristics) is **not** the same as modeling the logic value or decimal value of a 1-of-N signal. *Giramma* does not teach modeling a signal with a logic value that comprises an integer greater than 1.

Similarly, the Examiner apparently contends that the following statement from *Giramma*, Col. 1 lines 51-53 teaches the claimed signal strength element: "In addition to these state abstractions, some logic simulators use a 3-state model coupled with essentially unlimited strengths or values." Applicant does not dispute that there are 3-state signal models that model binary signals having various drive states. The 3-state model *Giramma* refers to is discussed in detail as background material in the instant specification. However, once again, this does not teach the functional limitations of the claimed signal strength of Applicant's model: it must comprise the drive state of a 1-of-N signal being modeled.

Finally, the Examiner asserts that because Giramma's model includes states to account for undefined signals at Col. 4, lines 53-55, that *Giramma* teaches the signal definition element in Claim 1. As before, *Giramma* does not teach the functional limitations of the claimed signal definition element: it must comprise the defined or undefined status of the nonbinary 1-of-N logic signal being modeled.

For these reasons, *Giramma* is not a proper §102(b) reference because it does not teach, suggest, or imply every element and every limitation of Claim 1, as amended.

IEEE Specification Std_logic_1164. The Examiner's rationale for rejecting claim 1 under this reference is essentially the same as his rationale for rejecting the claim under *Giramma*: the Examiner has confused the ability to model multiple states of a binary signal with the ability of the present invention to model multiple states of a non-binary, multiwire 1-of-N signal. The Examiner contends that the nine different signal states that the model handles shown in the reference at p. 15 (reproduced in the above discussion) teach the claimed signal value and the claimed signal definition. As before, these different signal state definitions simply do not teach a signal value that comprises the logic value of a nonbinary 1-of-N logic signal being modeled where the logic value further comprises an integer greater than 1. The signal values in the reference that are associated with signals having a value (i.e., signals that are not undefined, uninitialized, or high-impedance) are clearly limited to being 0 or 1.

The Examiner contends that the "strength stripper" functions mentioned in the reference at p. 15, Annex A that allegedly "transform 'Z', 'W', 'L', 'H', and '–' into their corresponding forcing strength counterparts" teaches the claimed signal strength element. In fact, these "strength stripper" functions essentially allow the simulation to "dumb down" the signal model by treating weakly driven signals having a value of 0 or 1 as strongly driven signals with the same value and either (1) converting everything else to "unknown;" or (2) propagating uninitialized signals and converting everything else to "unknown". This clearly does not teach the claimed signal strength element. Nevertheless, Applicant does not dispute that the signal model taught by the IEEE

reference includes some limited ability to account for the drive strength of the binary signal being modeled. However, because the reference does not teach the modeling of a nonbinary 1-of-N signal, the reference does not teach the claimed signal strength that is limited to being the drive state of a 1-of-N signal.

For these reasons, the IEEE reference is not a proper §102(b) reference because it does not teach, suggest, or imply every element and every limitation of Claim 1, as amended.

Rozon. The Examiner contends that *Rozon* teaches a signal model for an N-NARY logic simulation somewhere in the six pages starting on page 110 and ending on page 115. This comprises the entire article. The Examiner cites Table 1, page 110 as teaching a signal having a signal strength and a logic value greater than 1, and page 112 and Table 2 as teaching a signal definition.

The *Rozon* reference is a 4-page article with two pages of code fragments included as Annexes. *Rozon* explores how binary-based VHDL simulation techniques that account for multiple drive states of binary signals might be extended to non-binary (i.e., multi-valued) logic simulations and shows the simulation of a ternary (radix-3) full adder. However, other than showing how a ternary adder might be simulated, *Rozon* does not disclose or enable the simulation of other common functions that designers would use as the "building blocks" of a ternary (or other multi-valued) logic system. Consequently, *Rozon's* article does not disclose a signal model or simulation methodology for non-binary logic, rather, it is simply a very high level discussion of how one might go about developing these things ("establish an algebra", "choose a compiler", "decide on a description style," etc.) In fact, using Rozon's phraseology, the

signal model is the "description style" that is at issue here. *Rozon* informs us that a "description style" must be decided upon. Applicant respectfully submits that the claimed signal model is a "description style" that has already been "decided upon." Moreover, Applicant has decided upon a description style that, contrary to Rozon's assertion, accounts for signal drive strengths without increasing the model's complexity. See p. 111, col. 1.[4]

Rozon illustrates his approach to developing a simulation methodology by using it to model a ternary adder. Rozon models the adder using a multiplexer designated as a T-gate and a cycling gate (the "selected operators"), whose operations are characterized by equations (1) and (2) on p. 112. Rozon's ternary adder has three inputs: A, B, and a Carry In, any of which may have either a decimal value of 0, 1, or 2, or may be undefined, uninitialized, or high-impedance. At Annex 1, *Rozon* provides a code fragment that purportedly resolves whether these input signals have decimal values or are undefined, uninitialized, or high-impedance. Annex 2 is a code fragment that models the behavior of the adder, at the inputs' decimal value level, to predict the Sum and Carryout values of the adder. In other words, Rozon's signals have *either* a logic value (0, 1, or 2 in Rozon's ternary adder example) *or* the signals have no logic value but rather, are undefined, uninitialized, or high-impedance. *Notably, Rozon's signals do not have a drive strength.*

Turning to the Examiner's specific contentions, the Examiner asserts that Table 1, page 110 teaches a signal having a signal strength and a logic value greater than 1.

[4] *Rozon* opines that accounting for IEEE 1164 simulation extensions (such as the drive state of a signal) can only be done "at the cost of increasing the model's complexity." P. 111, first column. Applicant described this problem with Rozon's approach in the background discussion of the instant application at p. 8, line 12-18.

However, like Table 1 in the instant application, Rozon's Table 1 is background material; it is simply a repetition of the 9 states modeled in the MVL-9 model for binary logic described in IEEE_Std_logic_1164. As described above, any document that describes or teaches this well-known 9-state model for binary logic does not teach a signal model used in an N-NARY logic simulation that includes a signal value that comprises the logic value of a nonbinary 1-of-N logic signal being modeled where the logic value further comprises an integer greater than 1. Indeed, the entire point of Rozon's article is to speculate as to how one might go about adapting the MVL-9 model to apply it to signals having a signal value capable of being an integer greater than 1. For the same reason, Rozon's Table 1 does not teach a signal strength that comprises the drive state of a nonbinary 1-of-N logic signal being modeled.

The Examiner cites page 112 and Table 2 as teaching a signal definition. In fact, this table has nothing to do with a signal model that includes a signal definition that comprises the defined or undefined status of a nonbinary 1-of-N logic signal being modeled. Rather, this table lists the propagation delay of predesigned library cells that perform the logical function being simulated (in Rozon's example, a ternary adder) to enable designers to perform a timing analysis of their designs via simulation.

In sum, in his article, Rozon muses upon how one might approach adapting a standard extended state VHDL model intended to model and simulate binary logic to non-binary logic, and includes *one* example (the ternary adder) that does not include accounting for the drive strength of any of the input or output signals. Even if Rozon's article is very generously construed to disclose a methodology for simulating non-binary logic, Rozon does not disclose a way to model signals that can have *both* a logic value

that may be one of more than two decimal values *and* a drive strength. Accordingly, Rozon's article is not a proper §102(b) reference because it does not teach, suggest, or imply every element and every limitation of Claim 1, as amended.

Claim 2: *The model of Claim 1, wherein said logic value further comprises an integer less than or equal to 31.*

For the same reasons discussed above, the signal models disclosed by *Giramma*, IEEE_Std_logic_1164, and *Rozon* do not teach, suggest, or imply a signal model that includes all the claim elements and limitations of Claim 1 and models a nonbinary 1-of-N logic signal with a signal value comprising an integer greater than 1 and less than or equal to 31. However, the Examiner's specific contentions must be addressed.

The Examiner cites Giramma's col. 6, lines 51-54 ("Those skilled in the art will appreciate that 32-bit 'zoom' words, in accordance with the preferred embodiment of the invention, are used to obtain indices into various tables for next-state logic evaluations.") as support for his contention that, in addition to all the claim elements and limitations of Claim 1, *Giramma* teaches a signal model that models a nonbinary 1-of-N logic signal with a signal value comprising an integer greater than 1 and less than or equal to 31. However, as Applicant explained to the Examiner, Giramma's 32-bit "zoom" words have nothing to do with the logic value of a modeled signal; they are simply a convenient methodology to access the truth tables Giramma uses to determine a gate's output and send "directives" (i.e., inputs and state abstractions) to the next downstream gate. As the present invention is used in a simulation that does not utilize

truth tables, Giramma's 32-bit "zoom" words are wholly unrelated to any feature or function of the present invention.[5]

In response, the Examiner argues that the "zoom" words "read exactly on the Applicant's technology by abstracting the states of multiple logic gates in a "wire" bundle along with other signal attributes such as drive strength...". Final Office Action at 8. Moreover, the Examiner characterizes Giramma's technique of mapping the output of an 8-state gate to the input of a 4-state gate as

> "...mapping different characteristics of a "Bundle" of logic signals into one "zoom" word, which describes precisely what applicant is disclosing in the amended claims. Stating that the Giramma reference doesn't read on the applicant's technology because the "zoom" words are used to access a look up table ignores the essence of applicant's invention which is to abstract several signal wires into one 32 bit word and represent an "N-Nary" set of modeled logic signals, this is exactly what the Giramma reference is doing with the "zoom" words."

Id.

In fact, the Examiner is wrong on both counts. First, Giramma's technique does not map a "bundle of logic signals into one zoom word." Giramma maps the binary output of one primitive gate—say, an OR gate—from a signal model that may represent any one of 8 characteristic states of the output binary signal to a signal model that may represent any one of 4 characteristic states as the input for the next downstream primitive gate. This mapping occurs on each individual binary signal. Giramma discloses that the binary output signal may be in any one of the following 8 states: 0S (strongly driven low); 1S (strongly driven high); XS (strongly driven undefined); U

[5] As explained previously, *Giramma* employs an extended 8-state model only on certain gates in a logic path. For all other gates, *Giramma* maps certain states within the 8-state model to one of the states of the 4-state model, and then simulates the gates' outputs using the modified and mapped 4-state model. This technique enables the use of smaller truth tables to determine a mapped gate's output.

(uninitialized); 0R (weakly driven low); 1R (weakly driven high); XR (weakly driven undefined); and Z (high impedance). Col. 4, lines 1-2. Giramma's mapping technique redefines some of these states to be slightly different states in a 4-state model that supports the following states only: 0S, 1S, XS, and U. In converting any one signal, Giramma applies the following rules: weakly driven signals are mapped to strongly driven signals (e.g., 0R to 0S; 1R to 1S, and XR to XS); and high-impedance signals are mapped to strongly driven undefined (e.g. Z to XS). For each output signal, Giramma produces both an "8-state directive" and a "4-state directive" for the next downstream gate; if it is a gate in which a state that would otherwise be mapped would matter (e.g., a tri-state buffer, in which the Z state of the input signal is relevant) the signal is not mapped and an 8-state truth table is used to determine the gate's output. Alternatively, if it is not a gate that "cares" whether the input signal is strongly or weakly driven or high-impedance, then the signal is mapped and a much smaller 4-state truth table can be consulted to determine the gate's output. While Giramma does not describe or show the 32-bit zoom words in detail, he discloses that the words include bitfields that point to the base address of the relevant truth table(s) (which, of course, would be dictated by the type of gate being simulated), inputs, outputs, and offsets into the look-up tables to find next-state directives.

Next, the "essence" of Applicant's invention is not "to abstract several signal wires into one 32 bit word and represent an "N-Nary" set of modeled logic signals." Once again, this statement demonstrates a fundamental lack of understanding of Applicant's FAST14 technology. There is no abstraction; a 1-of-N signal used in Applicant's NDL logic is physically comprised of N wires (up to 32), its value is dictated

by which single wire of the N-wire bundle is asserted, and its validity depends upon whether one, and only one of the wires is asserted while all other wires in the N-wire bundle are not asserted. A 32-wire 1-of-N signal may have a logic value ranging anywhere from 0 to 31. There is no 32-bit "word" involved. Consequently, as described in the instant application, NDL gates are not limited to binary logic, but can be implemented having input and output signals in ternary, quaternary, or any other base logic (up to 32) that the designer finds convenient. For a more complete understanding of NDL logic, Applicant directs the Examiner's attention to U.S. Pat. No. 6,066,965, titled "Method and Apparatus for a N-Nary logic Circuit Using 1-of-4 Signals", which is one of the first patents obtained by Assignee that describes and claims Applicant's technology, and which is incorporated into the specification for the present invention. In addition, the patents listed in Appendix B disclose and claim various adders, floating point units, registers, RAM circuits, decoders, and a number of other building blocks and design tools that implement Assignee's well-defined FAST14 technology.

With respect to IEEE Std_logic_1164, the Examiner cites the description of the 9 states supported by the MVL-9 model shown on page 15, Annex A which appears in the text above as supporting his contention that the IEEE reference teaches a signal model having a logic value that further comprises an integer less than or equal to 31. As discussed previously, there is nothing in the MVL-9 model that supports modeling signals that are capable of having a logic value greater than 1. This model clearly does not support signals that are capable of having a logic value greater than 1, up to 31.

With respect to *Rozon*, the Examiner correctly cites that *Rozon* discloses a signal having a value comprising an integer less than or equal to 31 at Annex 1. As described

previously, *Rozon* discloses the simulation of a ternary adder, wherein the inputs may have a value of 0, 1, or 2. However, since *Rozon* does not disclose a signal that may have both a value of 0, 1, or 2 and a drive strength as required in Claim 1, *Rozon* does not anticipate Claim 2.

Accordingly, *Giramma*, IEEE_Std_logic_1164, and *Rozon* are not proper §102(b) references because they do not teach, suggest, or imply every element and every limitation of Claim 2, as amended.

Claim 3: *The model of Claim 1, wherein said signal strength further comprises one of the following: the high-impedance state, the weakly-driven state, the moderately-driven state, or the strongly-driven state.*

For the same reasons discussed above, the signal models disclosed by *Giramma*, IEEE_Std_logic_1164, and *Rozon* do not teach, suggest, or imply a signal model that includes all the claim elements and limitations of Claim 1 and models a nonbinary 1-of-N logic signal with a signal strength that comprises the high-impedance state, the weakly-driven state, the moderately-driven state, or the strongly-driven state. The Examiner points to various statements in *Giramma*, IEEE_Std_logic_1164, and *Rozon* as support for his contention that these references teach the claimed signal strength modeling capability. Applicant does not dispute that the signal models disclosed in *Giramma* and the IEEE reference teach modeling drive strengths of the binary signals that they model, but as discussed above, the references do not teach modeling a nonbinary 1-of-N logic signal having a signal value comprising an integer greater than 1 and a signal strength comprising the high-impedance state, the weakly-driven state, the moderately-driven state, or the strongly-driven state, as claimed in Claim 3. In Rozon's case, the reference to page 110 Table 1 is a reference to the MVL-

9 model discussed as introductory material in the article; as discussed above, *Rozon* does not teach the use of drive strength in his approach to adapting the MVL-9 model to nonbinary signals. Accordingly, *Giramma*, IEEE_Std_logic_1164, and *Rozon* are not proper §102(b) references because they do not teach, suggest, or imply every element and every limitation of Claim 3, as amended.

Nonstatutory Double Patenting Rejections

The Examiner rejected claims 1-3, 6-8, 11-13,16-18, and 21-23 under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1-7, 8-14, 15-21, 22-28, and 29-35 of Leight et al., US Patent No. 6,289,497, also assigned to Intrinsity, Inc. The Examiner states that "although the conflicting claims are not identical, they are not patentably distinct from each other because the claim in the Applicants' application is describing a signal naming convention that describes an N-nary logic circuit." Final Rejection at 9.

U.S. Pat. No. 6,289,497 discloses and claims a hardware description language that enables hardware designers designing NDL circuits to develop a design by describing its functions and interfaces using a syntax statement expressed in a novel version of ANSI C syntax referred to as "N-NARY C." The '497 patent is one of the patents incorporated by reference into the instant application on page 2 and referred to as "the Design Tools Patents." As described in the instant application at p. 14, lines 1-18 and FIG. 2, and disclosed in the '497 patent, the design tool compiles both a behavioral model and a physical circuit description from one syntax statement. The syntax statement further comprises one or more gate instantiations that comprise a gate

output signal variable defined according to the signal naming convention, a gate operator, and a gate expression that defines the logical function of the gate and the specific configuration of transistors required to build the gate.

As described in detail in the '497 patent, the signal naming convention imparts to the compiler components of the design tool information concerning the 1-of-N degree, evaluation, and clock phase of a gate's input and output signals. The 1-of-N degree field in each signal name corresponds to the number of individual wires (and thus the number of different values) that the named 1-of-N signal has. The evaluation field in each signal name indicates whether the gate driving the signal evaluates with a rising voltage (a "high-going" signal) or falling voltage (a "low-going" signal). Finally, the clock phase field in a signal name indicates which of the several overlapping clock phases that synchronize the logic in every NDL design during which the cell driving the signal evaluates. As described in the '497 patent, a signal name may also include a descriptor field and a bit field that allows designers specify the bit column to which the gate being constructed belongs to simplify floorplanning. In other words, the signal naming convention names the input and output signals of gates under design and describes their very broad parameters (width, high-going or low-going, and evaluation clock phase). Importantly, the signal naming convention described in the '497 patent is used only to generate schematics and the behavioral model; once generated, the behavioral model then simulates the behavior of the generated circuit using, as inputs, signals modeled in accordance with the present invention that are provided to the behavioral model by the user via the simulation environment files. Application at 15, lines 10-13.

FIG. 3 and page 14, line 19-p. 16, line 4 of the instant application describe the behavioral model generated by the Design Tool from the syntax statement. The behavioral model includes an input logic signal reader 30 that reads validity, value, and drive strength for each input signal from the simulation environment files according to the signal model of the present invention and provides that information to the arithmetic/logical operator 32, which outputs the value, drive strength, and validity of the gate's output signal(s) as a function of its input signal(s). The arithmetic/logical operator 32 provides that information to the output logic signal model generator 34, which then constructs a signal model according to the present invention that corresponds to each of gate's output signals.

As this summary demonstrates, the '497 patent describes a computer-aided design tool that enables hardware designers to automatically generate behavioral models and schematics by specifying the inputs, outputs, and desired functions of their designs in a software language similar to ANSI C. The signal naming convention described in the '497 patent defines only a signal's very broad, generic parameters, and is only used to develop gate instantiations, which in turn are compiled to generate schematics and behavioral models.

In contrast, the signal model of the present invention is an input to the compiled behavioral model. The signal model of the present invention defines a particular signal's specific characteristics such as the specific logic value, drive strength, and defined/undefined status at a particular point in time, for simulation in a behavioral model. Therefore, the signal model of the present invention does not have the necessary features of the signal naming convention used in N-NARY C (1-of-N degree,

evaluation voltage, and evaluation clock phase of a gate's input and output signals) and does have features that are absent from the signal naming convention (signal value, drive strength, and defined/undefined status). This is appropriate because the signal naming convention and the signal model are used for two very different purposes.

MPEP 804 provides that

> A double patenting rejection of the obvious-type is "analogous to [a failure to meet] the nonobviousness requirement of 35 U.S.C. 103" except that the patent principally underlying the double patenting rejection is not considered prior art. Therefore, any analysis employed in an obvious-type double patenting rejection parallels the guidelines for analysis of a 35 U.S.C. 103 obviousness determination. [Consequently,] the factual inquiries set forth in *Graham v. John Deere Co.* that are applied for establishing a background for determining obviousness under 35 U.S.C. 103 are employed when making an obvious-type double patenting analysis."

MPEP 804.II.B.1, citations omitted. Since the instant application was filed later than the application for the '497 patent, the appropriate determination is whether the invention defined in the claims of the instant application is an obvious variation of the invention defined in the claims of the '497 patent. MPEP 804.II.B.1.a

When performing a *Graham v. John Deere* analysis to assess obviousness under 35 USC § 103, MPEP 2141 defines the standards Examiners are to follow:

> When applying 35 USC 103, the following tenets of patent law must be adhered to:
>
> (A) The claimed invention must be considered as a whole;
>
> (B) The references must be considered as a whole and must suggest the desirability and thus the obviousness of making the combination;
>
> (C) The references must be viewed without the benefit of impermissible hindsight vision afforded by the claimed invention and
>
> (D) Reasonable expectation of success is the standard with which obviousness is determined

In addition, as in a standard 35 U.S.C. §103 analysis, the Examiner bears the initial burden of producing a *prima facie* case of obviousness. MPEP 2142. MPEP 2143 provides that

> To establish a prima facie case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves

> or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations.

Applying these standards, it is clear that the Examiner erred in concluding that the claims in the instant application are an obvious variation of the claims in the '497 patent. First, the Examiner failed to consider the claimed invention as a whole. Instead, the Examiner impermissibly distilled the invention down to its "thrust" or its "gist"—erroneously characterizing the claimed invention as a "signal naming convention that describes an N-nary logic circuit." Final Office Action at 9. Distilling an invention down to its "gist" or its "thrust" disregards the requirement of analyzing the subject matter "as a whole." MPEP 2141.02.

Moreover, one skilled in the art would not have modified the signal naming convention disclosed in the '497 patent by eliminating the 1-of-N degree field, the evaluation field, and the clock phase field, because the N-NARY design tool requires that information to produce the circuit schematic and behavioral model. Without these pieces of information, the tool would not "know" the width (i.e., how many conductors) of any specified signal, whether it is a signal internal to a gate or a signal that connects gates, or what phase clock the gate that drives the specified signal needs. In short, the signal naming convention would no longer function in its intended purpose, and the design tool that uses the signal naming convention would be unable to produce a schematic or behavioral model. The signal value, signal strength, and signal definition of the present invention signal model impart different information, because the signal model of the present invention is used for a different purpose. Indeed, signal value, signal strength, and signal definition information would be meaningless to the Design

Tool. These elements of the present invention are not analogous to the 1-of-N degree field, the evaluation field, and the clock phase field of the signal naming convention disclosed in the '497 patent, and a signal model according to the present invention would not function with the design tool to produce a circuit schematic and behavioral model because vital information would be missing.

Finally, as is evident by now, the '497 claims, combined with the information known by those of ordinary skill in the art, do not teach or suggest all the claim elements and limitations in Claims 1, 2, and 3 of the instant application. The '497 claims teach a 1-of-N degree field that "tells" the design tool how many conductors a specified signal has, they do not teach a signal value that comprises the logic value of a nonbinary 1-of-N logic signal being modeled, that further comprises an integer greater than 1 and less than 31. The '497 claims do not teach a signal strength that comprises the drive state of a nonbinary 1-of-N logic signal being modeled, and the '497 claims do not teach a signal definition that comprises the defined or undefined status of a nonbinary 1-of-N logic signal being modeled.

Accordingly, the claimed signal model is patentably distinct from the signal naming convention used in gate instantiations expressed in N-NARY C, and the Examiner's rejection of claims 1-3, 6-8, 11-13,16-18, and 21-23 under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1-7, 8-14, 15-21, 22-28, and 29-35 of Leight et al., US Patent No. 6,289,497 is improper.

Summary

In view of the above, Applicant believes that the Examiner's rejections of claims 1-3, 6-8, 11-13,16-18, and 21-23 were erroneous and respectfully requests a reversal of the Examiner's decision.

Respectfully submitted,

Karen S. Wright

Date: 06/04/2003

Karen S. Wright
Reg. No. 45,240

Karen S. Wright
Matthew J. Booth
Booth & Wright, L.L.P.
PO Box 50010
Austin, Texas 78763-0010
Telephone: (512) 474-8488
Facsimile: (512) 474-7996
mailto:karen.wright@boothlaw.com
mailto:matthew.booth@boothlaw.com
http://www.boothlaw.com/

Appendix A
Currently Pending Claims

1. CLAIMS

We claim the following invention:

1. A signal model used in an N-NARY logic simulation, comprising:

a signal value, said signal value further comprises the logic value of a nonbinary 1-of-N logic signal being modeled, wherein said logic value further comprises an integer greater than 1;

a signal strength, said signal strength further comprises the drive state of said nonbinary 1-of-N logic signal being modeled; and

a signal definition, said signal definition further comprises the defined or undefined status of said nonbinary 1-of-N logic signal being modeled.

2. The model of Claim 1, wherein said logic value further comprises an integer less than or equal to 31.

3. The model of Claim 1, wherein said signal strength further comprises one of the following: the high-impedance state, the weakly-driven state, the moderately-driven state, or the strongly-driven state.

4. Canceled.

5. Canceled.

6. A method that makes a signal model used in an N-NARY logic simulation, comprising:

assigning a signal value, said signal value further comprises the logic value of a nonbinary 1-of-N logic signal being modeled, wherein said logic value further comprises an integer greater than 1;

assigning a signal strength, said signal strength further comprises the drive state of said nonbinary 1-of-N logic signal being modeled; and

assigning a signal definition, said signal definition further comprises the defined or undefined status of said nonbinary 1-of-N logic signal being modeled.

7. The method of Claim 6, wherein said logic value further comprises an integer less than or equal to 31.

8. The method of Claim 6, wherein said signal strength further comprises one of the following: the high-impedance state, the weakly-driven state, the moderately-driven state, or the strongly-driven state.

9. Canceled.

10. Canceled.

11. A method that uses a signal model used in an N-NARY logic simulation, comprising:

reading a signal value, said signal value further comprises the logic value of a nonbinary 1-of-N logic signal being modeled, wherein said logic value further comprises an integer greater than 1;

reading a signal strength, said signal strength further comprises the drive state of said nonbinary 1-of-N logic signal being modeled;

reading a signal definition, said signal definition further comprises the defined or undefined status of said nonbinary 1-of-N logic signal being modeled; and

providing said signal value, said signal state, and said signal definition to the software-implemented simulation of the N-NARY logic design.

12. The method of Claim 11, wherein said logic value further comprises an integer less than or equal to 31.

13. The method of Claim 11, wherein said signal strength further comprises one of the following: the high-impedance state, the weakly-driven state, the moderately-driven state, or the strongly-driven state.

14. Canceled.

15. Canceled.

16. A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform a method that uses a signal model used in an N-NARY logic simulation, comprising:

reading a signal value, said signal value further comprises the logic value of a nonbinary 1-of-N logic signal being modeled, wherein said logic value further comprises an integer greater than 1;

reading a signal strength, said signal strength further comprises the drive state of said nonbinary 1-of-N logic signal being modeled;

reading a signal definition, said signal definition further comprises the defined or undefined status of said nonbinary 1-of-N logic signal being modeled; and

providing said signal value, said signal state, and said signal definition to the software-implemented simulation of the N-NARY logic design.

17. The program storage device of Claim 16, wherein said logic value further comprises an integer less than or equal to 31.

18. The program storage device of Claim 16, wherein said signal strength further comprises one of the following: the high-impedance state, the weakly-driven state, the moderately-driven state, or the strongly-driven state.

19. Canceled.

20. Canceled.

21. A signal modeling system used in an N-NARY logic simulation, comprising:

a signal value, said signal value further comprises the logic value of a nonbinary 1-of-N logic signal being modeled, wherein said logic value further comprises an integer greater than 1;

a signal strength, said signal strength further comprises the drive state of said nonbinary 1-of-N logic signal being modeled; and

a signal definition, said signal definition further comprises the defined or undefined status of said nonbinary 1-of-N logic signal being modeled.

22. The system of Claim 21, wherein said logic value further comprises an integer less than or equal to 31.

23. The system of Claim 21, wherein said signal strength further comprises one of the following: the high-impedance state, the weakly-driven state, the moderately-driven state, or the strongly-driven state.

24. Canceled.

25. Canceled.

Appendix B
Assignee's Patents Incorporating N-NARY or 1 of N Technology
As of 29 May 2003

	PAT. NO.	Title
1	6,571,378	Method and apparatus for a N-NARY logic circuit using capacitance isolation
2	6,567,835	Method and apparatus for a 5:2 carry-save-adder (CSA)
3	6,557,021	Rounding anticipator for floating point operations
4	6,460,134	Method and apparatus for a late pipeline enhanced floating point unit
5	6,445,213	Method for calculating dynamic logic block propagation delay targets using time borrowing
6	6,429,795	Method and apparatus for transforming pseudorandom binary patterns into test stimulus patterns appropriate for circuits having 1 of N encoded inputs
7	6,415,405	Method and apparatus for scan of synchronized dynamic logic using embedded scan gates
8	6,412,085	Method and apparatus for a special stress mode for N-NARY logic that initializes the logic into a functionally illegal state
9	6,404,233	Method and apparatus for logic circuit transition detection
10	6,367,065	Method and apparatus for N-Nary logic circuit design tool with precharge circuit evaluation
11	6,349,387	Dynamic adjustment of the clock rate in logic circuits
12	6,347,327	Method and apparatus for N-nary incrementor
13	6,345,381	Method and apparatus for a logic circuit design tool
14	6,334,136	Dynamic 3-level partial result merge adder
15	6,324,239	Method and apparatus for a 1 of 4 shifter
16	6,301,600	Method and apparatus for dynamic partitionable saturating adder/subtractor
17	6,301,597	Method and apparatus for saturation in an N-NARY adder/subtractor
18	6,295,622	Method and apparatus for transforming pseudorandom binary test patterns into test stimulus patterns appropriate for circuits having 1 of N encoded inputs
19	6,289,497	Method and apparatus for N-NARY hardware description language
20	6,288,589	Method and apparatus for generating clock signals
21	6,275,841	1-of-4 multiplier
22	6,275,838	Method and apparatus for an enhanced floating point unit with graphics and integer capabilities
23	6,272,514	Method and apparatus for interruption of carry propagation on partition boundaries
24	6,271,683	Dynamic logic scan gate method and apparatus
25	6,269,387	Method and apparatus for 3-stage 32-bit adder/subtractor
26	6,268,746	Method and apparatus for logic synchronization
27	6,252,425	Method and apparatus for an N-NARY logic circuit

28	6,233,707	Method and apparatus that allows the logic state of a logic gate to be tested when stopping or starting the logic gate's clock
29	6,223,199	Method and apparatus for an N-NARY HPG gate
30	6,219,687	Method and apparatus for an N-nary Sum/HPG gate
31	6,219,686	Method and apparatus for an N-NARY sum/HPG adder/subtractor gate
32	6,216,147	Method and apparatus for an N-nary magnitude comparator
33	6,216,146	Method and apparatus for an N-nary adder gate
34	6,211,456	Method and apparatus for routing 1 of 4 signals
35	6,209,076	Method and apparatus for two-stage address generation
36	6,202,194	Method and apparatus for routing 1 of N signals
37	6,181,596	Method and apparatus for a RAM circuit having N-Nary output interface
38	6,154,120	Method and apparatus for an N-nary equality comparator
39	6,124,735	Method and apparatus for a N-nary logic circuit using capacitance isolation
40	6,118,716	Method and apparatus for an address triggered RAM circuit
41	6,118,304	Method and apparatus for logic synchronization
42	6,115,294	Method and apparatus for multi-bit register cell
43	6,107,835	Method and apparatus for a logic circuit with constant power consumption
44	6,104,642	Method and apparatus for 1 of 4 register file design
45	6,088,830	Method and apparatus for logic circuit speed detection
46	6,069,836	Method and apparatus for a RAM circuit having N-nary word line generation
47	6,069,497	Method and apparatus for a N-nary logic circuit using 1 of N signals
48	6,066,965	Method and apparatus for a N-nary logic circuit using 1 of 4 signals
49	6,046,931	Method and apparatus for a RAM circuit having N-nary output interface